Adaptive Step-size Control in Simulation of diffusive CVD Processes

Jürgen Geiser and Christian Fleck

Humboldt-Universität zu Berlin,
Department of Mathematiks,
Unter den Linden 6, D-10099 Berlin, Germany
geiser@mathematik.hu-berlin.de

Abstract. In this paper, we present control strategies of a diffusion process for chemical vapor deposition for metallic bipolar plates.
In the models, we discuss the application of different models to simulate the plasma-transport of chemical reactants in the gas-chamber. The contribution are an optimal control problem based on a PID control to obtain an homogeneous layering.
We taken into account one- and two-dimensional problems, that are given with constraints and control functions. A finite element formulation with adaptive feedback control for time-step selection has been developed for the diffusion process. The optimization is presented with efficient algorithms.
Numerical experiments are discussed with respect to the diffusion processes of the macro-scopic model.

Keywords: Chemical vapor deposition, multi-scale problem, diffusion equations, PID control.
AMS subject classifications. 35K25, 35K20, 74S10, 70G65.

1 Introduction

We motivate our studying on simulating a low-temperature low-pressure plasma that can be found in CVD (chemical vapor deposition) processes. In the last years, due to the research in producing high temperature films by depositing of low pressure processes have increased. The interest on standard applications to TiN and TiC are immense but recently also deposition with new material classes known as MAX-phases are important. The MAX-phase are nanolayered terniar metal-carbides or -nitrids, where M is a transition metal, A is an A-group element (e.g. Al, Ga, In, Si, etc.) and X is C (carbon) or N (nitrid).

We present a model for low temperature and low pressure plasma, that can be used to implant or deposit thin layers of important materials. The applications of the implantation of the so called MAX-phases are used in the production of metallic bipolar plates, where the new metal must be non-corrosion and a good metallic conductor.

We present different model of the implantation process. First the process in the plasma-reactor, that transport the contaminants to the wafer-surface. We deal with a continuous flow model, while we assume a vacuum and a diffusion dominated process. Second the process at the wafer-surface is modeled by a heavy particles problem with underlying drift. This model deals more with the atomic behavior and we do not allow $p = 0$.

To solve the model equations we use analytical as also numerical methods. The analytical solutions are based on the Fourier- or Laplace-Transformation methods and are not longer discussed.

Numerical methods are described in the context of time and spatial discretization methods.

For the simulations we apply analytical as also numerical methods to obtain results to predict the grow of thin layers.

The paper is outlined as follows.

In Section 2 we present our mathematical model and a possible reduced model for the further approximations.

In Section 3 we discuss the theoretical background for the simulation of CVD processes.

The optimal control with the PID control approach is discussed in Section 4. The software and program-tools are discussed in Section 5. The numerical experiments are given in Section 6.

In the contents, that are given in Section 7, we summarize our results.

2 Mathematical Model

In the following, the models are discussed in two directions of far-field and near-field problems:

1. Reaction-diffusion equations, see [9] (far-field problem);
2. Boltzmann-Lattice equations, see [22] (near-field problem).

The modeling is considered by the Knudsen Number (Kn), which is the ratio of the mean free path λ over the typical domain size L. For small Knudsen Numbers $Kn \approx 0.01 - 1.0$, we deal with a Navier-Stokes equation or with the Convection-Diffusion equation, see [15] and [20], whereas for large Knudsen Numbers $Kn \geq 1.0$, we deal with a Boltzmann equation, see [21].

2.1 Modeling with partial differential equations

Dynamic processes with modifications in time and and space will be reshaped by partial differential equations. There is 1.) the pde-formula itself which describes the physical laws of nature that influence the process and 2.) initial and boundary conditions in which specific characteristics of the process, like boundary behavior, can be coded.

There are two types of boundary conditions, namely Dirichlet and Neumann boundary. With the Dirichlet type the exact value of the boundary is known

however with Neumann boundaries the time derivation of the boundary values in normal direction is known.

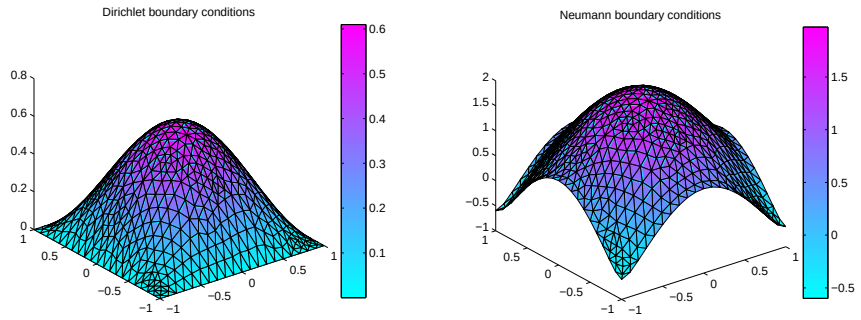


Fig. 1. Dirichlet and Neumann boundary conditions.

2.2 Model for Optimal control of the Layer

We will concentrate us on a continuum model of mass transportation and assume that the energy and momentum is conserved, [Gobbert1996]. Therefore the continuum flow of the mass can be described as convection-diffusion reaction equation given as:

$$\partial_t c - \nabla D \nabla c - R_g = 0, \text{ in } \Omega \times [0,T] \tag{1}$$

$$c(x,0) = c_0(x), \text{ on } \Omega, \tag{2}$$

$$\frac{\partial c(x,t)}{\partial n} = c_1(x,t), \text{ on } \partial\Omega \times [0,T], \tag{3}$$

where c is the molar concentration. D is the diffusion and v is the velocity. R_g is the reaction term between the different concentrations.

We modify our model equation 1 to a control problem with an additionally right-hand side source:

$$\partial_t c - \nabla D \nabla c = c_{source}, \text{ in } \Omega \times [0,T] \tag{4}$$

$$c(x,0) = c_0(x), \text{ on } \Omega, \tag{5}$$

$$\frac{\partial c(x,t)}{\partial n} = c_1(x,t), \text{ on } \partial\Omega \times [0,T], \tag{6}$$

where $c_{source}(x,t)$ is a discontinuous or continuous source flow of the concentration c.

We assume an optimal concentration at the layer :
$c_{opt}(x,t)$ where the layer is given as $x \in \Omega_{layer}$
and our constraints are given as :

$c_{source,min} \leq c_{source} \leq c_{source,max}$
Additionally, we have to solve the minimization problem:

$$\min\left(J(c, c_{source})\right) := 1/2 \int_T \int_{\Omega_{layer}} |c(x,t) - c_{opt}(x,t)|^2 dxdt$$
$$+\lambda/2 \int_T \int_{\Omega} |c_{source}(x,t)|^2 dxdt, \tag{7}$$

where T is the time period of the process.

Remark 1. We choose the L_2-error to control our minimization problem. In the literature, see [25] and [19], there exists further control-errors, which respect the time-behavior.

In a first work we only solve the transport equation with UG and try to find out the optimal control of the sources to obtain the best homogeneous layer.

In a second work we consider the optimal control problem and solve also the backward problem.

3 Theoretical Background for Simulation of diffusive CVD Processes

In the following we discuss the approximation methods and errors for the simulation of the CVD processes.

3.1 Approximation and Discretization

For the numerical solutions we need to apply approximation methods, e.g. finite difference methods and iterative solver methods for the nonlinear differential equations.

The finite element discretization is based on Ω_h the variational boundary value problem reduces to find $u_h \in V_h$ satisfying the initial condition $u_h(0)$ such that

$$\int_{\Omega_h} \Big(\frac{\partial u_h}{\partial t} v_h + D\nabla u_h \cdot \nabla v_h \Big) dx = 0, \tag{8}$$
$$\text{for all } v_h \in V_h, \tag{9}$$

We define the minimal length of triangle which we get from the spatial discretization with Δx.

This leads to the following linear semidiscretized system of ordinary differential equations :

$$M \frac{du^*}{dt} + Au^* = 0, \tag{10}$$

where M is the mass and A the M-matrix.

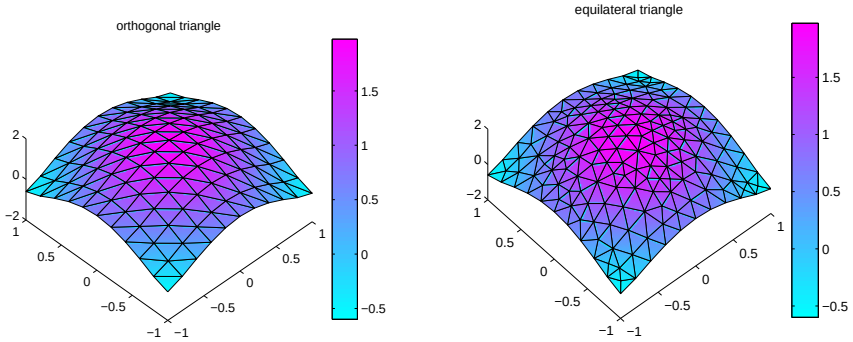


Fig. 2. Spatial discretization.

Here we have taken into account the CFL-condition (Courant-Friedrichs-Levy), which is given as

$$CFL = 2D_{max}\frac{\Delta t}{\min_{e\in E}\Delta x_e^2}, \tag{11}$$

where D_{max} is the maximal diffusion parameter, I is the set of the edges of the discretization. We restrict the CFL-condition to 1, if we use an explicit time-discretization and can lower the condition, if we use an implicit discretization.

For the explicit time discretization, we apply explicit Euler or Runge-Kutta methods

We use the explicit lower order Runge-Kutta methods:

$$\begin{array}{c|cc} 0 & & \\ \frac{1}{2} & \frac{1}{2} & \\ \hline & 0 & 1 \end{array} \tag{12}$$

Furthermore we use the following Heun method (third order) :

$$\begin{array}{c|ccc} 0 & & & \\ \frac{1}{3} & \frac{1}{3} & & \\ \frac{2}{3} & 0 & \frac{2}{3} & 0 \\ \hline & \frac{1}{4} & 0 & \frac{3}{4} \end{array} \tag{13}$$

The implicit time discretization is done with implicit Euler or Runge-Kutta methods.

Here, we use the implicit trapezoidal rule:

$$\begin{array}{c|cc} 0 & & \\ 1 & \frac{1}{2} & \frac{1}{2} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array} \tag{14}$$

Furthermore we use the following Gauss Runge-Kutta method :

$$\begin{array}{c|cc} \frac{1}{2}-\frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4}-\frac{\sqrt{3}}{6} \\ \frac{1}{2}+\frac{\sqrt{3}}{6} & \frac{1}{4}+\frac{\sqrt{3}}{6} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array} \tag{15}$$

Remark 2. We apply implicit time-discretization methods for the pure diffusion part, where we apply explicit time-discretization methods for the pure convection part. Here we have to respect the CLF-condition, see [12].

3.2 Errors and convergence rate

For studying the errors and the convergence-rates in our test example we have to define the following norm in two space-dimensions:

– Discrete L_{max}-norm :

$$err_{L_{max},\Delta x,\Delta t} = \max_{i=1}^{p} |c_{num}(x_i,T) - c_{ref}(x_i,T)|, \tag{16}$$

– Discrete L_1-norm :

$$err_{L_1,\Delta x,\Delta t} = \sum_{i=1}^{p} \Delta x^2 |c_{num}(x_i,T) - c_{ref}(x_i,T)|, \tag{17}$$

– Discrete L_2-norm :

$$err_{L_2,\Delta x,\Delta t} = \sqrt{\sum_{i=1}^{p} \Delta x^2 |c_{num}(x_i,T) - c_{ref}(x_i,T)|^2}, \tag{18}$$

where Δx is the spatial step of the discretization, Δt is the time step of the discretization and T is the end-time of the computation. c_{num} is the numerical solution and c_{ref} is the reference solution, computed at fine spatial- and time-grids.

The numerical convergence rate are given as:

– For the spatial error, we define:

$$\rho_{L_2,\Delta x_1,\Delta x_2,\Delta t} = \frac{log(\frac{err_{L_2,\Delta x_1,\Delta t}}{err_{L_2,\Delta x_2,\Delta t}})}{log(\frac{\Delta x_1}{\Delta x_2})}, \tag{19}$$

where Δx_1 is the coarse and Δx_2 is the fine spatial grid-step and Δt is the time grid step for both results.

– For the time error, we define:

$$\rho_{L_2,\Delta x,\Delta t_1,\Delta t_2} = \frac{log(\frac{err_{L_2,\Delta x,\Delta t_1}}{err_{L_2,\Delta x,\Delta t_2}})}{log(\frac{\Delta t_1}{\Delta t_2})}, \tag{20}$$

where Δt_1 is the coarse and Δt_2 is the fine time-step and Δx is the spatial grid step for both results.

We often use $\Delta x_2 = \frac{\Delta x_1}{2}$. In this case, we have $\rho_{L_2,\Delta x_1,\Delta x_2,\Delta t} = \rho_{L_2,\Delta x_1,\Delta t}$. Furthermore we will have for fix Δx_1 attention to the $\Delta t \in I = [0, \Delta t_{max}]$, which have maximal ρ. Thus we define $ArgMax(\Delta x)$:

$$ArgMax(\Delta x) := \arg\max_{\Delta t \in I} \rho_{L_2,\Delta x,\frac{\Delta x}{2},\Delta t}. \tag{21}$$

4 Optimal Control Methods

Here we discuss the control of a diffusion equation with a feedback based on a PID-controller.

4.1 Forward Controller (simple P-Controller)

The first controller we discuss is the simple P-controller, see [23]. A first idea to control linearly the error of the solved PDE.

In the Figure 3 we present the P-controller.

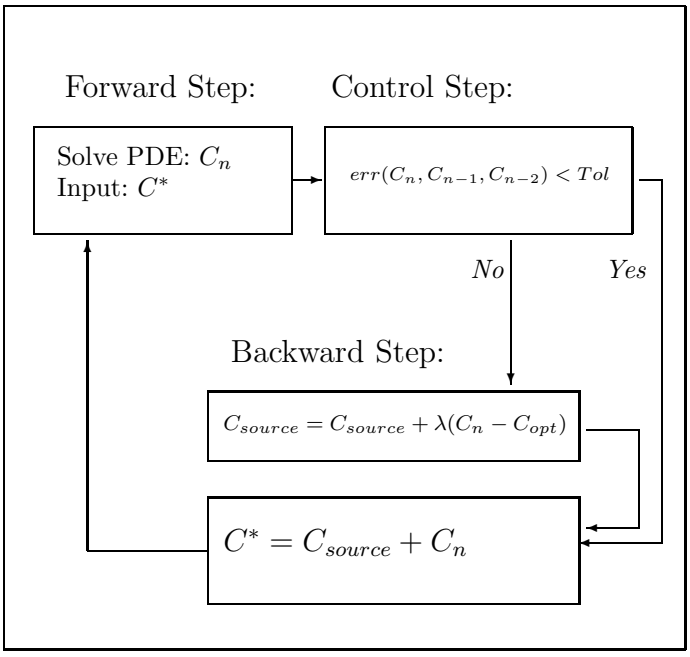


Fig. 3. P-controller for the solution C.

Our control problem is given with the control of the error to the optimal concentration of the layer and correct the source-flux.

$$\partial_t c + v\nabla c - \nabla D\nabla c = c_{source}, \text{ in } \Omega \times [0,T] \tag{22}$$

$$c(x,0) = c_0(x), \text{ on } \Omega, \tag{23}$$

$$\frac{\partial c(x,t)}{\partial n} = c_1(x,t), \text{ on } \partial\Omega \times [0,T], \tag{24}$$

where $c_{source}(x,t)$ is a discontinuous or continuous source flow of the concentration c.

We assume an optimal concentration at the layer :
$c_{opt}(x,t)$ where the layer is given as $x \in \Omega_{layer}$
and our constraints are given as :
$c_{source,min} \leq c_{source} \leq c_{source,max}$

Remark 3. Taken into account the hysteresis of the deposition process, we apply a linear increase of the optimal control with respect to time, see Figure 4.

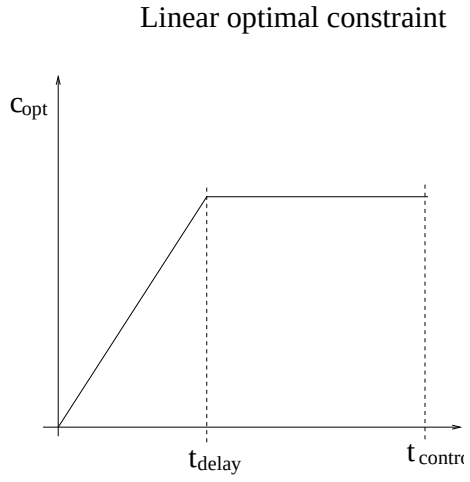


Fig. 4. Linear constraint c_{opt} for the deposition process.

4.2 PID-Controller

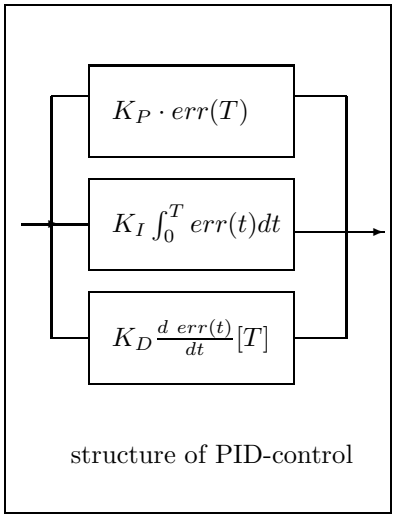
The PID controller is used for controlling temperature, motion, flow and its available in analog and digital forms, see [23]. The controller helps to get the output (velocity, temperature, position) in the area of the constraint output, in a short time, with minimal overshoot, and with small errors. In many applications the PID controller help to control the output.

We have three elements in the control:
P -Proportional, I - Integral, D - Derivative.

These terms describe three basic mathematical functions applied to the error signal , $error = C_{optimal} - C_{computed}$. This error represents the difference between the constraint (optimal set) and the computed results in the simulation. The controller performs the PID mathematical functions on the error and applies the their sum to a process (motor, heater, etc.).

To tune a system means adjusting three multipliers K_P, K_I and K_D adding in various amounts of these functions to get the system to behave the way you want, see [23].

The table below summarizes the PID terms and their effect on a control system.



structure of PID-control

Effect on Control System: The main influence in a control loop, K_P reduces a large part of the overall error. reduces the final error in a system. Summing even a small error over time produces a drive signal large enough to move the system toward a smaller error. Counteracts the K_P and K_I terms when the output changes quickly. This helps reduce overshoot and ringing.

Table 1. PID-Control.

Initialization of the PID-controller:

The algorithm of the initialization of the PID-control, i.e. search K_P, K_I, K_D, is given as :

Initialization of the PID-controller, see [25] with the following algorithm:

Algorithm 1 *1.) We initialism the P-controller :* $K_I = 0.0, K_D = 0.0$.

2.) The amplifying factor K_P *is increased till we reached the permanent oscillations as a stability boundary of the closed control system.*

3.) We obtain for K_P the critical value $K_{P,crit.}$.
4.) The period-length of the permanent oscillation is given as T_{crit}.
5.) We obtain the next parameters with :

Controller	K_P	T_n	T_v
P	$0.5K_{P,crit.}$		
PI	$0.45K_{P,crit.}$	$0.85T_{crit}$	
PID	$0.6K_{P,crit.}$	$0.5T_{crit}$	$0.12T_{crit}$

Table 2. Heuristic derivation of the control-parameters.

Further we compute the rest parameters as. $K_I = K_P/T_n$, $K_D = K_P/T_v$, *see [16].*

4.3 Adaptive Time-control

Often the heuristic assumptions of the PID-parameters are to coarse.

One can improve the method by applying an adaptive step-size control.

We discuss the step-size control with respect to our underlying error, that is given by the computed and optimal output of our differential equation.

Based on the adaptive control we can benefit to accelerate the control problem.

According to Hairer and Wanner [11], we apply the automatic control problem with a PID controller.

The automatically step-size is given as, see [23].

$$\Delta t_{n+1} = \left(\frac{e_{n-1}}{e_n}\right)^{K_P} \left(\frac{tol}{e_n}\right)^{K_I} \left(\frac{e_{n-1}^2}{e_n e_{n-2}}\right)^{K_D} \Delta t_n, \tag{25}$$

where *tol* is the tolerance, e_n is the error of the quantities of interest in time step Δt.

We can control the step-size with respect to our heuristically computed K_P, K_I and K_D parameters.

Initialization of the adaptive control:

Algorithm 2 *1.) Define Tolerance, Min and Max of the concentration*
2.) Apply the parameters : K_P, K_D, K_I form a first run
3.) Optimize the computations with a first feedback.

5 Software and Program-Tools

Matlab-functions We use the Matlab-toolbox for the time- and spatial discretizations, where we have a finite element method with $P1$-elements and an implicit Euler method for the time-discretization.

matlab function	**description**	**enhancement**
parabolic	solve parabolic pde (heat equation)	In the future we plan to use academical code to compute the pde-solutions (e.g. θ-method). Furthermore we will use Cosmol/Femlab.
pdetool	matlab toolbox to create the geometry of the FEM-structure and the boundary conditions	We have also implemented an alternative mesh, with orthogonal triangles. Matlab always uses equilateral triangles.
refine	This function refines the geometry of the mesh. All triangles, were replaced with four new triangles.	For convergence rate calculations we must guarantee that the geometry near the points, which are changed in the backward step (source), are similar.
guide	matlab toolbox to create graphical user interfaces (gui)	

DEPOSIT-PID toolbox The PID-controller is also programmed in Matlab. Our combined code is given in the DEPOSIT-PID toolbox and described in the following. The DEPOSIT-PID toolbox is manipulated by a graphical user interface, by which simulation- and control-models are chosen and corresponding parameters can be manually adjusted.

The objective is to simulate the diffusion and deposition of the vapor in the apparatus and to obtain the optimal vapor concentration at the measuring point.

We can divide the process into three phases, namely forward step, control step, backward step, that have a cyclic repetition. In the forward step the diffusion takes place, which can be simulated by a time-step of the heat equation.

After that, in the control step, the actual concentration at the under boundary $(0, -1)$ can be measured (shown by computed in the graph) and compared with the optimal value (shown by optimal).

The control step is followed by the backward step: From the error in control step and the control model the optimal alteration of the source can be computed. The vapor flows through the source point $(0, 0)$ in the apparatus and this value is shown by the SourceOutput in the graph. This will be simulated through

the addition of SourceOutput to actual concentration at the source point (see following figure). Further, as a simplification, we set the concentration at the under boundary to zero, because it is here that the gas transforms into solid matter.

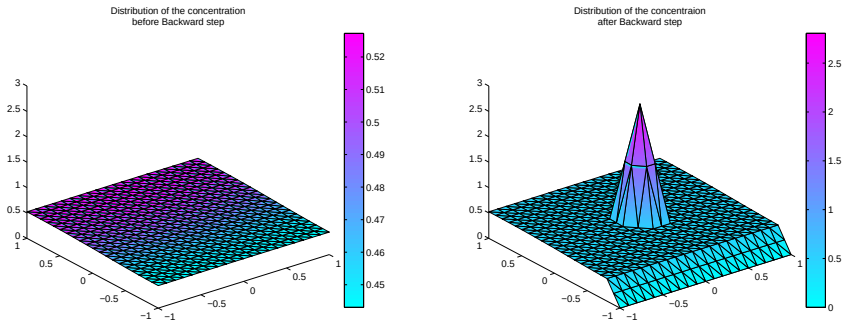


Fig. 5. Backward step.

In Section 4 we introduced some fundamentals in order to control the apparatus. The models and corresponding parameters can be altered by the Gui.

The Layout of the DEPOSIT-PID gui

1 Short time plot (2D) of computed, optimal and Source Output
2 Long time plot (2D)
3 Listbox with names parameters
3a Textbox with actual value of the parameter chosen in [3]
3b Textbox to change the value of the parameter chosen in [3]
4 Listbox with names parameters
4a Textbox with actual value of the parameter chosen in [4]
4b Textbox to change the value of the parameter chosen in [4]
5 3D plot of distribution
6 3D grid plot of distribution
7 Listbox with names of parameters
7a Checkbox with actual value of the parameter chosen in [7]
8 Push button: Save
9 Push button: Reset
10 Radio button: Start

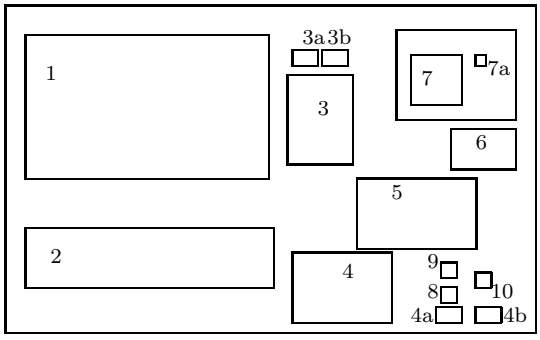
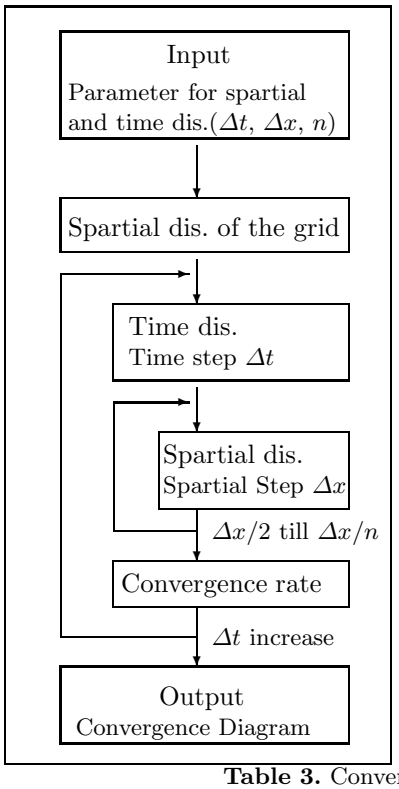


Fig. 6. Layout of the DEPOSIT-PID Gui .

KONTOOL Another software-tool KONTOOL is programmed to compute the numerical convergence rates of the applications. The software-tool has implemented the errors and convergence rates defined in Section 3.2. An error-analysis based on successive refinement of space and time is done and the resulting errors and convergence rates are computed. Optimal convergence rates with respect to balance the time- and spatial-grids are calculated, see Algorithm 3.

Remark 4. The software-tool can be modified and applied to arbitrary spatial- and time-discretization methods. The interface of KONTOOL needs at least the parameters of the spatial and time grid and the starting parameters of the underlying methods.



The algorithm of the computation of the numerical convergence tableau :

Algorithm 3 *1.) We compute reference solutions. a.) numerically : fine time and spatial steps or b.) analytically (if there exists an analytical solution)*
2.) We apply one spatial discretization of step Δx and apply all time discretization with steps Δt, where the coarsest Δt is given by the CFL condition or till first non-numerical results as oscillations. We compute the error $c_{num} - c_{ref}$ in the L_2-norm.
3.) We continue the next fine spatial steps, e.g. $\Delta x/2$.
4.) We compute the convergence tableau with time and space.

Table 3. Convergence Diagram Tool.

In the next section we discuss the numerical experiments.

6 Experiment for the plasma reactor

In this section, we present our numerical experiments for the CVD processes in a plasma reactor.

6.1 Simulation of a diffusion equation with analytical solution (Neumann boundary conditions)

Here we simulate a diffusion equation with Neumann boundary conditions and right hand side 0. Our control problem has only the forward problem to solve and we consider the accuracy of our simulations.

We have the following equation :

$$\partial_t c - \beta^2(\partial_{xx} + \partial_{yy})c = 0, \text{ in } \Omega \times [0,T] \tag{26}$$

$$c(x,y,0) = \cos(2x) + \cos(2y), \text{ on } \Omega, \tag{27}$$

$$\partial_n c(x,y,t) = 0, \text{ on } \partial\Omega \times [0,T], \tag{28}$$

where c is the molar concentration. $\Omega = [-1,1] \times [-1,1]$ and $t \in (0,T)$. $D = \beta^2$ is the diffusion parameter of the diffusion equation and $f(t)$ is the right hand side or diffusion source.

We have the following analytical solution :

$$c_{ana}(x,y,t) = \sin(2) + \sum_{n=1}^{\infty} A_n \exp(-\beta^2 n^2 \pi^2 t)(\cos(n\pi x) + \cos(n\pi y)), \quad (29)$$

where $A_n = (-1)^n \frac{-4\sin(2)}{n^2\pi^2 - 4}$. We apply the diffusion coefficient $D = 0.01$ resp. $\beta = 0.1$. We obtain the following result after time $T = 12.0$, see Figure 7:

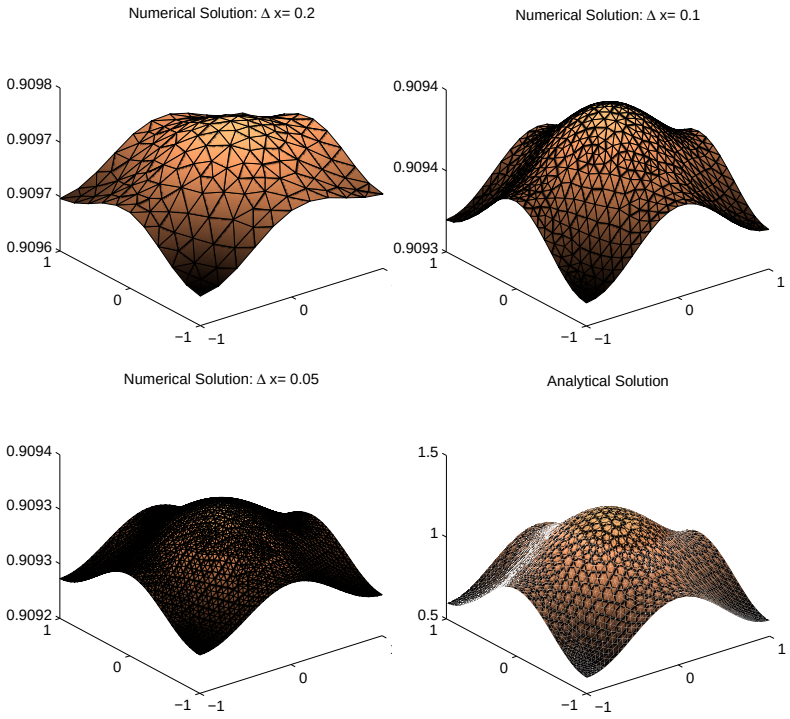


Fig. 7. 2D experiment of the diffusion equation at the end-time $T = 12.0$, $\beta = 0.1$.

We see that for large Δx the numerical solution converge faster to the stable constant endsolution (offset) than solution for smaller Δx and the analytical solution. The error between the constant analytical endsolution (offset, $sin(2) = 0.9093$) is also greater than for smaller Δx.

The L_2-error is given in the following Table 4

Δx	offset(analy.)	max(num.)	min(num.)	L_1-error	L_2-error
0.20	0.90**92**97426826	0.90**21**08407643	0.90**21**08381295	$7.189e-3$	$5.17e-5$
0.10	0.90**92**97426826	0.90**75**14994036	0.90**75**14987008	$1.782e-3$	$3.18e-6$
0.05	0.90**92**97426826	0.90**88**58511500	0.90**88**58509666	$4.389e-4$	$1.93e-7$

Table 4. Offset convergence ($\Delta t = 0.1$, $Timestep = 100$, $\beta = 0.61644$).

Remark 5. We test for the pure diffusion equation our underlying discretization methods and apply finite elements for the spatial discretization and implicit Runge-Kutta methods for the time discretization. In the results we obtain decreasing errors for the different time- and spatial steps.

6.2 Simulation of an optimal control of a diffusion equation with heuristic choise of the control parameters

Here we simulate a first example of a diffusion equation and control the concentrations in the deposition process.

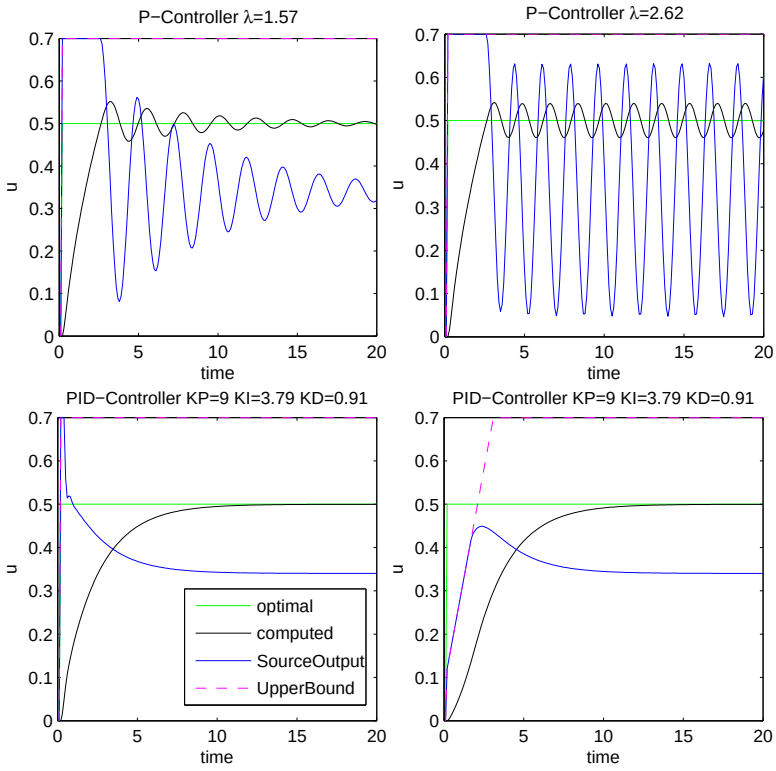


Fig. 8. 2D experiment of the diffusion equation and control of a single point.

We have the following equation :

$$\partial_t c - \nabla D \nabla c = f(t), \text{ in } \Omega \times [0,T], \tag{30}$$

$$c(x,0) = c_0(x), \text{ on } \Omega, \tag{31}$$

$$\frac{\partial c(x,t)}{\partial n} = c_1(x,t), \text{ on } \partial\Omega \times [0,T], \tag{32}$$

where c is the molar concentration. D is the diffusion parameter of the diffusion equation and $f(t)$ is the right hand side or diffusion source. We have the following constraint : $c_{optimal}(x_{point}, y_{point}) = 0.5$, where (x_{point}, y_{point}) is the control point in our domain. The parameters are given as : $D = 0.01$ and $f(t) = at + b$,

so we deal with a linear source, $a = 0.2$ and $b = 0.1$ are constants. In the next tests we propose the 3 possibilities to control the optimal temperature

- P-Control with constant source
- PID-Control with constant source
- PID-Control with Linear source

The results with for the control methods are given in Figure 8.

6.3 Preliminary remark for simulations for the convergence order

To determine the function $ArgMax$, firstly we have to determine a practicable interval I for

$$ArgMax(\Delta x) := \arg\max_{\Delta t \in I} \rho_{L_2, \Delta x, \frac{\Delta x}{2}, \Delta t}. \tag{33}$$

One possibility is the Interval

$$I_{CFL} := (0, \Delta t_{CLF}], \tag{34}$$

with $\Delta t_{CFL} := \frac{\Delta s^2}{2D_{max}}$ from the CFL-condition. In the experiments we find another interval, where the convergence rate function is relative stable convex:

$$I_{stable} := (0, \Delta t_{stable}], \tag{35}$$

where

$$\Delta t_{stable} := \max_{\Delta t} \{\rho : (0, \Delta t) \to \mathbb{R} \; is \; convex\}. \tag{36}$$

It is clear that we can in I_{stable} take some restrictions to a subinterval $I_{sub} := [\Delta t_{smin}, \Delta t_{smax}] \subset I_{stable}$, when we know that is some $\Delta t \in I_{sub}$ with $\rho(\Delta t) > \rho(\Delta t_{smin}), \rho(\Delta t_{smax})$.

The function of the numerical convergence rate is discrete in the spatial discretization variable Δx since we get a finer discretization with a bisection of Δx. With a finer discretization, a triangle is replaced by four sub-triangles.

In the temporal discretization variable Δt we are in contrast not restricted to such conditions and Δx could be chosen to any arbitrary value above 0. To get a first glance, we have selected the methods of bisection. Subsequently, we consider finer discretizations in intervals which are of special interest.

6.4 Simulations for the convergence order

We consider the convergence rate ρ as a function in dependence of Δx and Δt. This function also depends on some parameters for instance the diffusion coefficient D and the control parameter λ. We now present the results of two chosen experiments. For the first experiment the parameters are $D = 0.1$ and $\lambda = 1$, while we increase the propagation velocity in such a way that we increase D to 1 for the second experiment.

Δx	Δt	err $= \lvert u_{num,\Delta x,\Delta t} - u_{num,fine\Delta x,\Delta t} \rvert$	Convergence Rate
0.1	0.1	0.077007	3.2531
0.1	0.05	0.016153	4.0077
0.1	0.025	0.0020085	3.9447
0.1	0.0125	0.00026087	1.8276
0.1	0.00625	0.00014699	0
0.05	0.1	0.27873	2.8591
0.05	0.05	0.076833	3.7481
0.05	0.025	0.011437	4.052
0.05	0.0125	0.001379	3.7776
0.05	0.00625	0.00020111	0
0.025	0.1	0.6564	2.4552
0.025	0.05	0.23939	3.2449
0.025	0.025	0.050505	4.008
0.025	0.0125	0.0062781	3.9999
0.025	0.00625	0.00078482	0
0.0125	0.1	1.04	2.1424
0.0125	0.05	0.4711	2.7173
0.0125	0.025	0.14327	3.7251
0.0125	0.0125	0.021668	4.0516
0.0125	0.00625	0.0026133	0
0.00625	0.1	1.2092	1.586
0.00625	0.05	0.80554	2.4179
0.00625	0.025	0.30148	3.2179
0.00625	0.0125	0.064803	4.005
0.00625	0.00625	0.0080726	0

Table 5. Numerical results for the P-controller for different spatial steps with $D = 0.1$ and $\lambda = 1.0$ as P-value for the controller.

We observe that for instance in the case of spatial discretization $\Delta x = 0.05$ and $x = 0.25$ the maximal convergence rate lies between $\Delta t = 0.05$ and $\Delta t = 0.0125$. The maximum itself lies close to 0.025. To determine the precise value of $\Delta t = ArgMax(\Delta x)$ we refine our method in these interesting intervals.

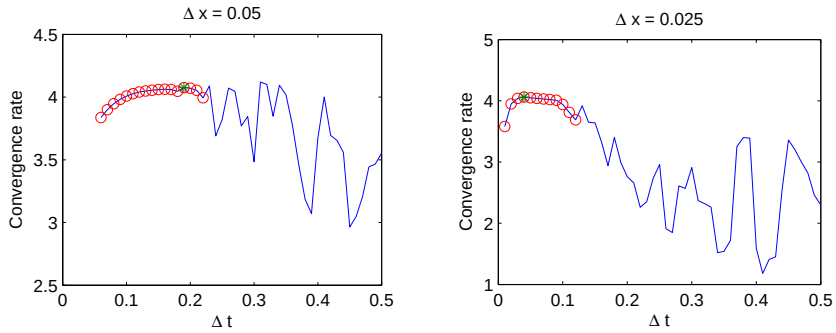
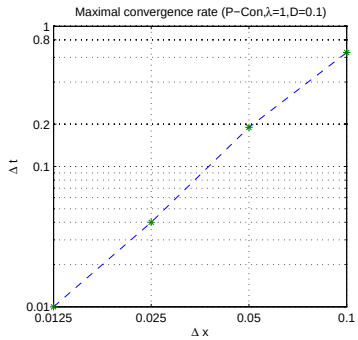


Fig. 9. ρ for a P-Controller, D=0.1.

In the plot we observe that there is a relatively stable convex area starting at 0. This area then switches over to an area with strong fluctuations. We now compare the stable area with the CFL-condition. The area where $CFL < 1$ lies in the stable convex area of the function ($\Delta t_{CFL} < \Delta t_{stable}$). For $\Delta x = 0.05, D = 0.1$ the CFL-area ends at $\Delta t_{CFL} = 0.0125$, the stable convex area range to about $\Delta t_{stable} = 0.2$. For our convergence diagram we try to find for every Δx the value Δt where the convergence rate becomes maximal (ArgMax). It is clear that we only use Δt at which the convergence rate function is stable: $\Delta t < \Delta t_{stable}$.



We can see in the loglog-plot $ArgMax(\Delta x)$ linear dependence.

Δx	$ArgMax$	$-log(Max)$
.10000	.65000	0.62
.05000	.19000	2.4
.02500	0.04	4.64
.01250	.01000	6.64

This table show for $\Delta x = 0.1$, 0.05, 0.025, 0.0125 the associated Δt, which have maximal convergence rate ($ArgMax$).

Table 6. Convergence Diagram (KONTOOL).

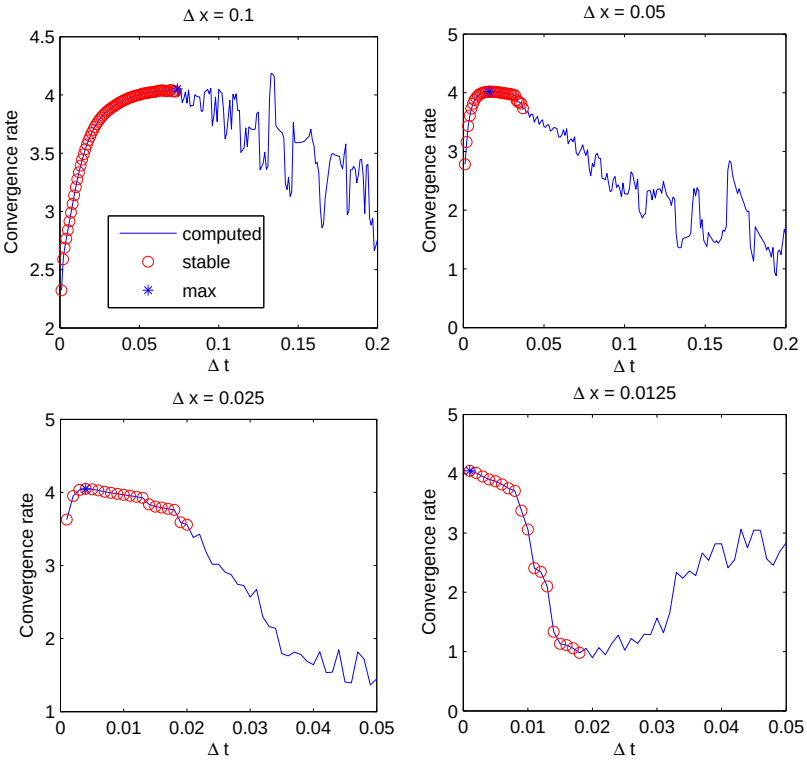
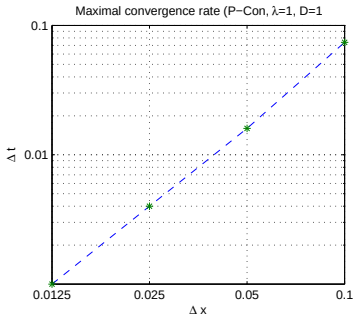


Fig. 10. ρ for a P-Controller, D=1.



The loglog-plot $ArgMax(\Delta x)$ with linear dependence.

Δx	$ArgMax$	$-log(Max)$
.10000	0.0700	3.84
.05000	0.0170	5.88
.02500	0.0040	7.97
.01250	0.0010	9.97

This table show for $\Delta x = 0.1$, 0.05, 0.025, 0.0125 the associated Δt, which have maximal convergence rate ($ArgMax$).

Table 7. Convergence Diagram (KONTOOL).

Remark 6. The experiment shows the linear convergence rate of the P-controller with different λ values. So we obtain a stable method with respect to the P-controller. In the examples, we apply heuristic methods to derive the control parameters for the P- and PID-controller. We show that we have reached the linear order of the underlying finite element discretization method. We have higher control errors if we did not compute the correct control parameters and the numerical errors are smaller, than our control error. To prohibut this problem we have to compute in the next example the control parameters by a feedback equation, see [24].

6.5 Simulation of an optimal control of a diffusion equation with adaptive control

In the second example we simulate the diffusion equation and control the temperature with and adaptive control based on a PID controller, see [23].

We have the following equation :

$$\partial_t c - \nabla D \nabla c = f(t), \text{ in } \Omega \times [0,T] \tag{37}$$

$$c(x,t) = c_0(x), \text{ on } \Omega, \tag{38}$$

$$\frac{\partial c(x,t)}{\partial n} = c_1(x,t), \text{ on } \partial\Omega \times [0,T], \tag{39}$$

where c is the molar concentration. D is the diffusion parameter of the diffusion equation and $f(t)$ is the right hand side or source.

We have the following constraint :

$c_{optimal}(x_{point}, y_{point}) = 0.5$

where $(x_{point}, y_{point}) = (0, -1)$ is the control point in our domain.

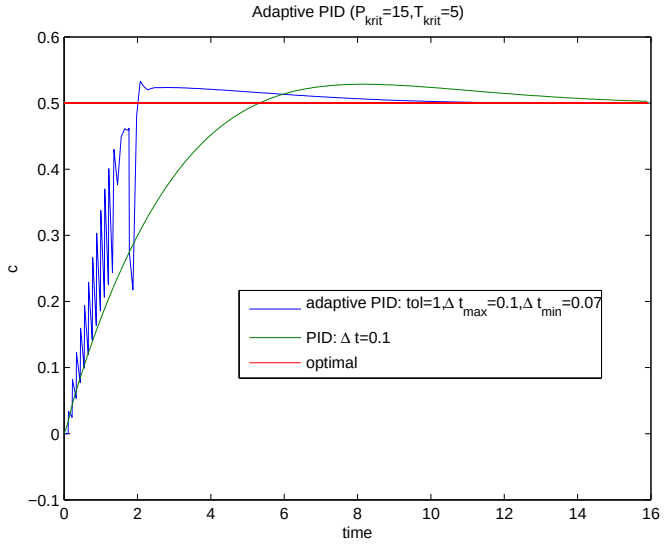


Fig. 11. 2D experiment with and without the adaptive time-step control.

The automatically step-size is given as, see [23].

$$\Delta t_{n+1} = \left(\frac{e_{n-1}}{e_n}\right)^{K_P} \left(\frac{tol}{e_n}\right)^{K_I} \left(\frac{e_{n-1}^2}{e_n e_{n-2}}\right)^{K_D} \Delta t_n, \tag{40}$$

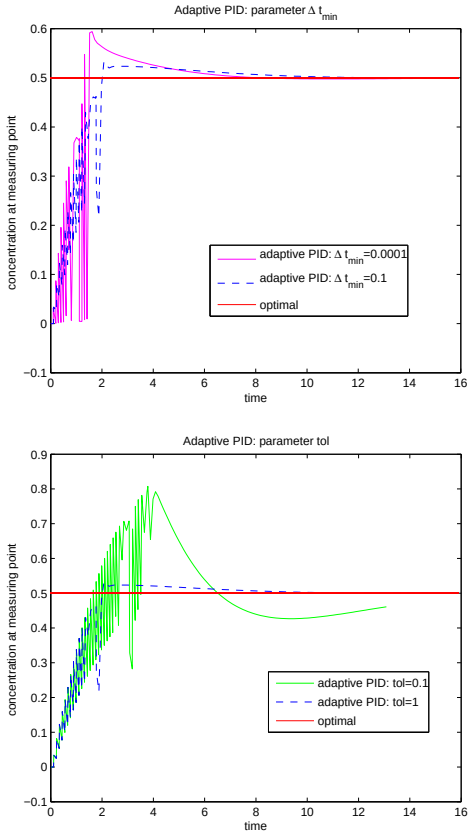


Fig. 12. Adaptive PID with modified parameters.

where $tol = 1$ is the tolerance, e_n is the error of the quantities of interest in time step Δt.
The errors are given as

$$e_n = \frac{||u_n - u_{n-1}||}{||u_n||}, \tag{41}$$

where u_n is the result at time-step t^n.

The parameters are given as : $D = 0.1$, $P_{krit} = 15$, $T_{krit} = 5$, Δt (PID-Control) , $tol = 1$ (adaptive PID-Control) $\Delta t_{max} = 0.1$ (adaptive PID-Control), $\Delta t_{min} = 0.01$ (adaptive PID-Control) (see Figure 11).

Furthermore we change the parameter $tol = 1$ to $tol = 0.1$ and $\Delta t_{min} = 0.01$ to $\Delta t_{min} = 0.0001$ (see Figure 12).

Remark 7. In the Figures 11 and 12, we see an oscillating time interval at the beginning of the automatically step-size control. In the first experiments, we did

only taken into account a previous heuristically computation of the control parameters K_P, K_I and K_D before the step-size control for the whole time-interval $0, T$. The optimal control parameters are given to the whole time-interval $0, T$. A modified algorithm to compute the control parameters for the initialization time-interval $0, t_{begin}$ and the whole time-interval $0, T$ overcome the oscillation problems.

A modified automatically step-size control, which minimize the oscillations is given in the following Algorithm 4.

Algorithm 4 *1.) We compute the reference control parameters $K_{P,global}$, $K_{I,global}$ and $K_{D,global}$ for the time-interval $0, T$.*

2.) We apply the automatically step-size control for the global control parameters with tol_{global}, $\Delta t_{max,global}$ and $\Delta t_{min,global}$, which can by chosen large.

3.) We stop the computation till we reach the optimal solution and mark remember the time t_{oszill}.

4.) We compute the local control parameters $K_{P,local}$, $K_{I,local}$ and $K_{D,local}$ for the time-interval $0, t_{oszill}$.

5.) We restart the computation with the local control parameters and smaller step-size parameters tol_{local}, $\Delta t_{max,local}$ and $\Delta t_{min,locall}$ till we reach t_{oszill} and continue the computation with the global parameters.

6.) We stop the computation if we reach $t = T$. If we obtain also high oscillation with the local parameters, we refine the local interval and goto step 3.).

Remark 8. The modified automatically step-size control taken into account the local behavior of the control problem. We could adapt the control parameters K_P, K_I and K_D with respect to the local time-intervals. This modified algorithm considers a local time behavior more accurate and reduces oscillations at the initialization process.

7 Conclusions and Discussions

We present a continuous or kinetic model, due to the fare field or near field effect of our deposition process. We discuss the PID-controller to automatize our deposition process. Due to heuristic methods of deriving the PID parameters, we discuss an aposteriori error estimates to automatize the time-stepping methods. A modified automatically step-size control is discussed and will be considered for future applications. For the mesoscopic scale model we discussed different experiments and their convergence rates. Further numerical examples are presented to discuss the influence of near-continuum regime at the thin film. In future, we will analyze the validity of the models with physical experiments.

References

1. I. Farago, and Agnes Havasi. *On the convergence and local splitting error of different splitting schemes.* Eötvös Lorand University, Budapest, 2004.

2. P. Csomós, I. Faragó and A. Havasi. *Weighted sequential splittings and their analysis.* Comput. Math. Appl., (to appear)
3. K.-J. Engel, R. Nagel, *One-Parameter Semigroups for Linear Evolution Equations.* Springer, New York, 2000.
4. I. Farago. *Splitting methods for abstract Cauchy problems.* Lect. Notes Comp.Sci. 3401, Springer Verlag, Berlin, 2005, pp. 35-45
5. I. Farago, J. Geiser. *Iterative Operator-Splitting methods for Linear Problems.* Preprint No. 1043 of the Weierstrass Institute for Applied Analysis and Stochastics, Berlin, Germany, June 2005.
6. J. Geiser. *Numerical Simulation of a Model for Transport and Reaction of Radionuclides.* Proceedings of the Large Scale Scientific Computations of Engineering and Environmental Problems, Sozopol, Bulgaria, 2001.
7. J. Geiser. *Gekoppelte Diskretisierungsverfahren für Systeme von Konvektions-Dispersions-Diffusions-Reaktionsgleichungen.* Doktor-Arbeit, Universität Heidelberg, 2003.
8. J. Geiser. *Discretization methods with analytical solutions for convection-diffusion-dispersion-reaction-equations and applications.* Journal of Engineering Mathematics, published online, Oktober 2006.
9. M.K. Gobbert and C.A. Ringhofer. *An asymptotic analysis for a model of chemical vapor deposition on a microstructured surface.* SIAM Journal on Applied Mathematics, 58, 737–752, 1998.
10. Ch. Grossmann and H.-G. Ross. *Numerik partieller Differentialgleichungen.* Teubner Studienbücher, Mathematik, 1994.
11. E. Hairer and G. Wanner. *Solving Ordinary Differential Equatons II.* SCM, Springer-Verlag Berlin-Heidelberg-New York, No. 14, 1996.
12. W. Hundsdorfer and J.G. Verwer. *Numerical solution of time-dependent advection-diffusion-reaction equations*, Springer-Verag, Berlin, Heidelberg, New-York, 2003.
13. S. Karaa. *High-Order Compact ADI Methods for Parabolic Equations.* Journal of Computers and Mathematics with Applications, Vol. 52, Iss. 8-9, 1343–1356, 2006.
14. S. Karaa. *High-Order Difference Schemes for 2-d Elliptic and Parabolic Problems with Mixed Derivatives.* Wiley InterSciences, published online, October 2006.
15. H.H. Lee. *Fundamentals of Microelectronics Processing* McGraw-Hill, New York, 1990.
16. J. Lee and Th.F. Edgar. *Continuation Method for the Modified Ziegler-Nichols Tuning of Multiloop Control Systems.* Ind. Eng. Chem. Res., 44 (19), 7428 -7434, 2005.
17. M.A. Lieberman and A.J. Lichtenberg. *Principle of Plasma Discharges and Materials Processing.* Wiley-Interscience, John Wiley & Sons, Inc Publication, Second edition, 2005.
18. Chr. Lubich. *A variational splitting integrator for quantum molecular dynamics.* Report, 2003.
19. H. Lutz and W. Wendt. *Taschenbuch der Regelungstechnik.* Harri-Deutsch Verlag,Issue 6, Frankfurt, Germany, 2005.
20. S. Middleman and A.K. Hochberg. *Process Engineering Analysis in Semiconductor Device Fabrication* McGraw-Hill, New York, 1993.
21. M. Ohring. *Materials Science of Thin Films.* Academic Press, San Diego, New York, Boston, London, Second edition, 2002.
22. T.K. Senega and R.P. Brinkmann. *A multi-component transport model for non-equilibrium low-temperature low-pressure plasmas.* J. Phys. D: Appl.Phys., 39, 1606–1618, 2006.

23. A.M.P. Valli, G.F. Carey and A.L.G.A. Coutinho. *Control strategies for timestep selection in simulation of coupled viscous flow and heat transfer.* Communications in Numerical Methods in Engineering, 18:2, 131–139, 2002.
24. T. Yamaguchi and K. Shimizu. *Asymptotic Stabilitzation by PID Control: Stability Analysis Based on Minimum Phase and High-Gain Feedback.* Electrical Engineering in Japan, vol. 156, No. 1, 783–791, 2006.
25. J.G. Ziegler and N.B. Nichols. *Optimum settings for automatic controllers.* Trans. ASME, 64, 759-768, 1942.